

HowToSortPandasDataframe

September 8, 2025

```
[1]: import pandas as pd
```

```
[2]: data = {'Name': ['Alice', 'Bob', 'Charlie', 'David'],
           'Age': [25, 30, 35, 40],
           'Score': [85, 90, 95, 80]}
```

```
[3]: df = pd.DataFrame(data)
```

```
[4]: sorted_df = df.sort_values(by='Age')
```

```
[5]: print(sorted_df)
```

	Name	Age	Score
0	Alice	25	85
1	Bob	30	90
2	Charlie	35	95
3	David	40	80

```
[6]: data = {'Name': ['Alice', 'Bob', 'Charlie', 'David'], 'Age': [25, 30, 35, 40], 'Score': [85, 90, 95, 80]}
df = pd.DataFrame(data)
sorted_df = df.sort_values(by='Age', ascending=False)
print(sorted_df)
```

	Name	Age	Score
3	David	40	80
2	Charlie	35	95
1	Bob	30	90
0	Alice	25	85

```
[7]: data = {'Name': ['Alice', 'Bob', 'Charlie', 'David'],
           'Age': [25, 30, 35, 40],
           'Score': [85, 90, 95, 80]}
df = pd.DataFrame(data)
sorted_df = df.sort_values(by=['Age', 'Score'])
print(sorted_df)
```

	Name	Age	Score
0	Alice	25	85

```
1      Bob   30    90
2  Charlie   35    95
3    David   40    80
```

```
[8]: data_with_nan = {"Name": ["Alice", "Bob", "Charlie", "David"], "Age": [28, 22, None, 22]}
df_nan = pd.DataFrame(data_with_nan)
sorted_df = df_nan.sort_values(by="Age", na_position="first")
print(sorted_df)
```

```
      Name   Age
2  Charlie   NaN
1      Bob  22.0
3    David  22.0
0    Alice  28.0
```

```
[9]: data = {'Name': ['Alice', 'Bob', 'Charlie', 'David'],
            'Age': [25, 30, 35, 40],
            'Score': [85, 90, 95, 80]}
df = pd.DataFrame(data)
df_sorted_by_index = df.sort_index()
print(df_sorted_by_index)
```

```
      Name   Age  Score
0    Alice   25    85
1      Bob   30    90
2  Charlie   35    95
3    David   40    80
```

```
[10]: data = {'Name': ['Alice', 'Bob', 'Charlie', 'David'],
            'Age': [25, 30, 35, 40],
            'Score': [85, 90, 95, 80]}
df = pd.DataFrame(data)
df_sorted_by_index_desc = df.sort_index(ascending=False)
print(df_sorted_by_index_desc)
```

```
      Name   Age  Score
3    David   40    80
2  Charlie   35    95
1      Bob   30    90
0    Alice   25    85
```

```
[11]: data = {
        "Name": ["Alice", "Bob", "Charlie", "David", "Eve"],
        "Age": [28, 22, 25, 22, 28],
        "Score": [85, 90, 95, 80, 88]
}
df = pd.DataFrame(data)
```

```
sorted_df = df.sort_values(by='Age', kind='quicksort')
print(sorted_df)
```

	Name	Age	Score
1	Bob	22	90
3	David	22	80
2	Charlie	25	95
0	Alice	28	85
4	Eve	28	88

```
[12]: data = {
    "Name": ["Alice", "Bob", "Charlie", "David", "Eve"],
    "Age": [28, 22, 25, 22, 28],
    "Score": [85, 90, 95, 80, 88]
}
df = pd.DataFrame(data)

sorted_df = df.sort_values(by='Age', kind='mergesort')
print(sorted_df)
```

	Name	Age	Score
1	Bob	22	90
3	David	22	80
2	Charlie	25	95
0	Alice	28	85
4	Eve	28	88

```
[13]: data = {
    "Name": ["Alice", "Bob", "Charlie", "David", "Eve"],
    "Age": [28, 22, 25, 22, 28],
    "Score": [85, 90, 95, 80, 88]
}
df = pd.DataFrame(data)

sorted_df = df.sort_values(by='Age', kind='heapsort')
print(sorted_df)
```

	Name	Age	Score
1	Bob	22	90
3	David	22	80
2	Charlie	25	95
4	Eve	28	88
0	Alice	28	85

```
[14]: data = {
    "Name": ["Alice", "Bob", "Charlie", "David", "Eve"],
    "Age": [28, 22, 25, 22, 28],
    "Score": [85, 90, 95, 80, 88]
}
```

```
df = pd.DataFrame(data)

sorted_df = df.sort_values(by='Name', key=lambda col: col.str.lower())
print(sorted_df)
```

```
Name  Age  Score
0    Alice   28     85
1      Bob   22     90
2  Charlie   25     95
3    David   22     80
4      Eve   28     88
```

[]: